Start Up

Beijer ELECTRONICS

Programmable I/O (PIO) NA-9379, Communication

KI00342 2014-10

1 Function and area of use

The Programmable I/O (PIO) combines CODESYS control with the opportunity to build control systems to the exact size and specifications for the I/O signals involved.
In this document “PIO” are used for “Programmable I/O” (NA-9379).

2 About this Start Up document

This Start Up document should not be considered as a complete manual. It is an aid to be able to start up a normal application quickly and easily.

Use the following software and drivers in order to obtain a stable application:

Programming software

- CODESYS V3.5 SP3 Patch 1 or later , programming the PIO
- Compiler version 3.5.3.10 (CODESYS V3.5 SP3 Patch 1)
- IOGuidePro 1.1.0.8 #0004 or later

Device description, Crevis PIO NA-9379

- PIO_DeviceDescription 20140416.devdesc(.xml) or later, enable PIO in CODESYS

Library manager, CODESYS

- Standard 3.5.2.0 (System)
- IoStandard 3.5.3.0 (System)
- Time and Date 3.5.1.0
- Ethernet 3.4.2.0
- Modbus Master TCP 3.5.3.0 (IoDrvModbusTCP)
- Modbus Slave TCP 3.5.3.0 (IoDrvModbusTCPSlave)

Beijer Electronics Automation AB – *a Beijer Electronics Group company*

Head Office
Beijer Electronics Automation AB
P.O. Box 426, Stora Varvsgatan 13a
SE-201 24 Malmö, SWEDEN
Telephone +46 40 35 86 00
Fax +46 40 93 23 01

Subsidiaries
Norway, Drammen: Beijer Electronics AS, ☎ +47 32 24 30 00
Finland, Vantaa: Beijer Electronics Oy, ☎ +358 207 46 35 00
Denmark, Roskilde: Beijer Electronics A/S, ☎ +45 75 76 66
Estonia, Tallinn: Beijer Electronics Eesti Oü, ☎ +372 6 518140
Latvia, Riga: Beijer Electronics SIA, ☎ +371 6 7842280
Lithuania, Kaunas: Beijer Electronics UAB, ☎ +370 5 2323101

Reg no. 556701-3965 | VAT no SE556701396501 | www.beijer.se | info@beijer.se

KI_eng_130531.dot, 130103

In this document we have used following hardware and software

- PIO (NA-9379), ST-3424, ST-4422, ST-1218, ST-2328, ST-3702
- CODESYS V3.5 SP3 Patch 1
- IO Guide Pro 1.1.0.8 #4

For further information we refer to

- CODESYS, Manuals and help in the software
- NA-9379 Manual www.beijer.se
  - *NA-9379, User Manual*
  - *NA-9379, Specification Preliminary*
  - *NA-9379, Quick Guide*
- Crevis Manual www.beijer.se
  - *FnIO Configuration Parameter/Memory/Register*
  - *Crevis FnIO S-Series All manuals*

- Start Up document
  - "Programmable I/O (PIO) NA-9379, Basic", KI00341
  - "iX TxA/TxB - Programmable I/O (PIO) NA-9379", KI00343

This document and other Start Up documents can be obtained from our homepage.
Please use the address manual@beijer.se for feedback on our Start Up documents.

3 First step!

For best understanding and easy working with Programmable I/ O and CODESYS we recommend:
- Read Start Up document "Programmable I/O (PIO) NA-9379, Basic setting", KI00341
- Basic knowledge of CODESYS. There is a quick start in CODESYS help file.
- Prepare the computer with the programming tools CODESYS and IOGuidePro.
- Download manuals

Note!
- Always use the correct version of the drivers and software!
- When using BootP in IOGuidePro: use Windows XP (windows 7, 64 bit are not supported).

4 Table of Contents

5 Communication Modbus TCP/RTU

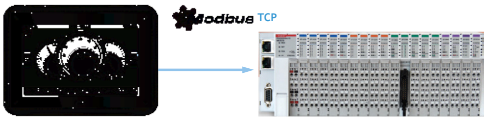
There are 18 socket connection for Modbus communication.

- 6 UDP: Is used by CODESYS, BootP, NA-9379 (Network Variable), etc
- 12 TCP: Is used by Modbus TCP Master/Slave.
  - Example NA-9189/NA-9289 (Modbus TCP Slave).
  - Six of this connection are "TCP_LISTEN" used by HMI, etc

Note!
The number of connected devices is depended on the data length, scan time, performance of PIO.
The socket connection will be operated with socket open and close. It means that the TCP socket will be support up to 12 pcs simultaneously.
- Its limitation is 6 pcs of NA-9189/NA-9289, but it is depended on the situations!

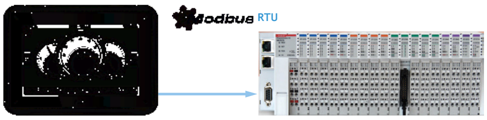
5.1 iX TxA/TxB (Modbus TCP Master) – PIO (Modbus TCP Slave)

The driver communication in CODESYS use a Function Block.
For more information see StartUp document
"iX TxA/TxB - Programmable I/O (PIO) NA-9379", KI00343



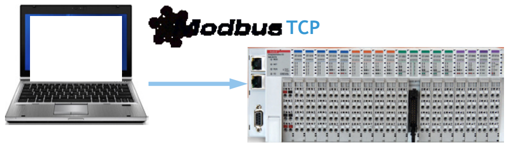
5.2 iX TxA/TxB (Modbus RTU Master) – PIO (Modbus RTU Slave)

Under construction



5.3 Modbus TCP Master – PIO (Modbus TCP slave)

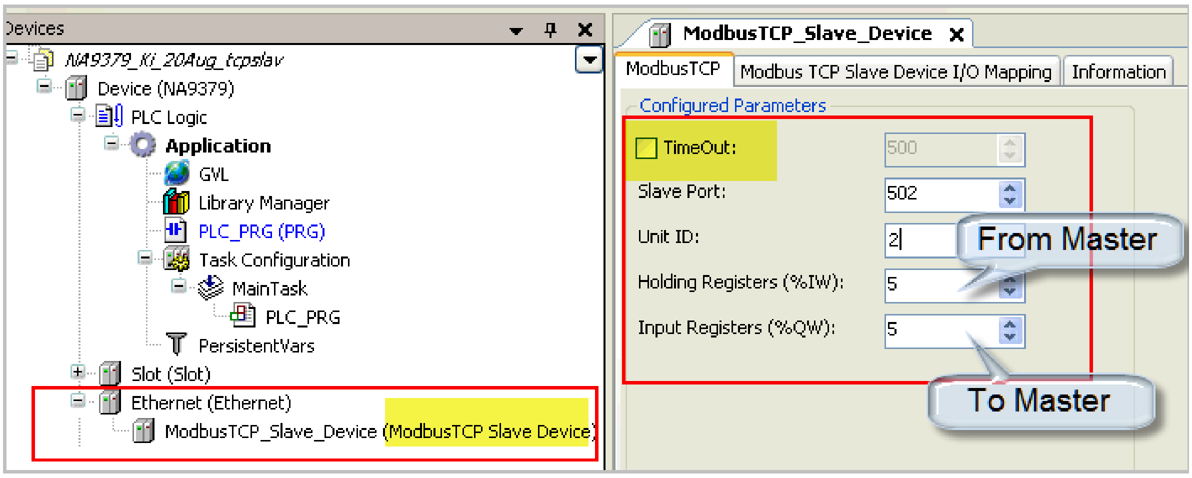
Communication from a Modbus TCP Master to PIO.



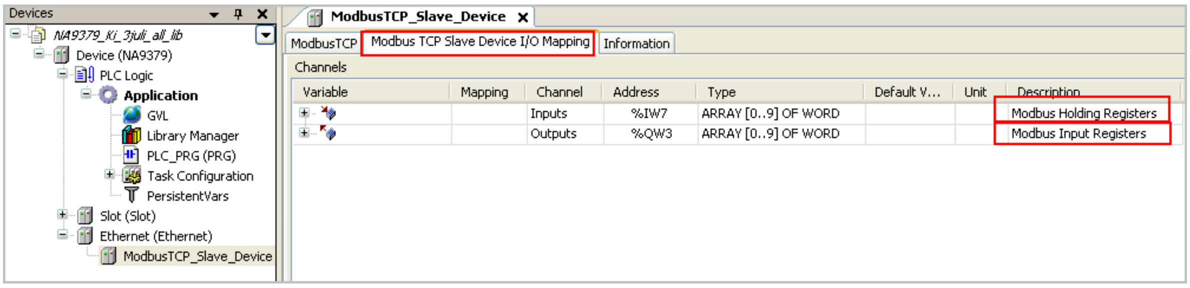
For more information, see help in CODESYS.

Note!

- Use functions code for Holding and Input register in master
- Holding register can only be read in CODESYS
- Information in help file: Max 40 bytes in/out (we have test with 100 register in/out and it works also)
- Link: Adresses Holding and Input Register



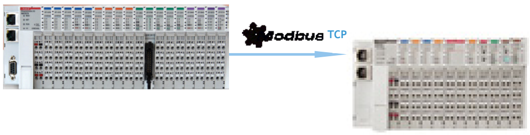
□ TimeOut:	500
Slave Port:	502
Unit ID:	2
Holding Registers (%IW):	5
Input Registers (%QW):	5



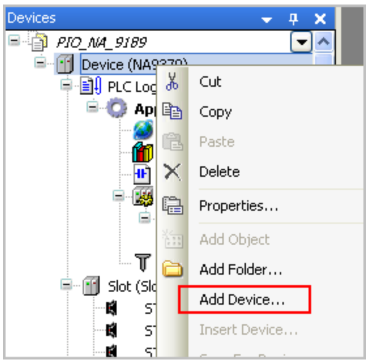
Variable	Mapping	Channel	Address	Type	Default V...	Unit	Description
		Inputs	%IW7	ARRAY [0..9] OF WORD			Modbus Holding Registers
		Outputs	%QW3	ARRAY [0..9] OF WORD			Modbus Input Registers

5.4 PIO Modbus TCP Master – NA-9189/-9289 Modbus TCP Slave

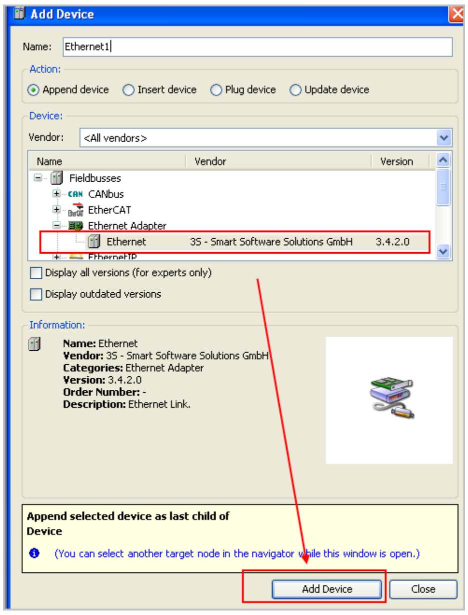
Communication to Modbus TCP Slave (NA-9289)



- Add Device
  - Click Device (NA9379), right click, Click 'Add Device'



- Add "Ethernet Adapter"
  - Click "Fieldbusses", 'Ethernet Adapter / 'Ethernet', 'Add Device'



Name: Ethernet1

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

Device:

Vendor: <All vendors>

Name	Vendor	Version
Fieldbusses		
CANbus		
EtherCAT		
Ethernet Adapter		
Ethernet	3S - Smart Software Solutions GmbH	3.4.2.0
EthernetIP		

□ Display all versions (for experts only)

□ Display outdated versions

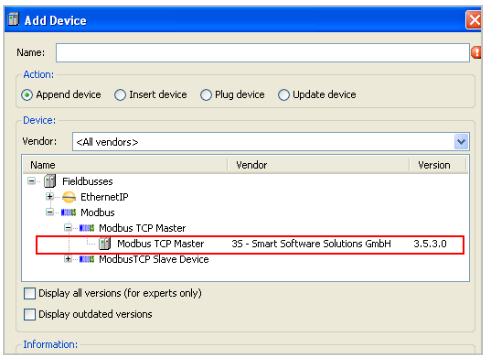
Information:

Name: Ethernet
Vendor: 3S - Smart Software Solutions GmbH
Categories: Ethernet Adapter
Version: 3.4.2.0
Order Number: -
Description: Ethernet Link.

Append selected device as last child of Device

(You can select another target node in the navigator while this window is open.)

- Add "MODBUS TCP Master"
  - Modbus TCP Master / "Modbus TCP Master"
  - 'Add Device'



Name:

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

Device:

Vendor: <All vendors>

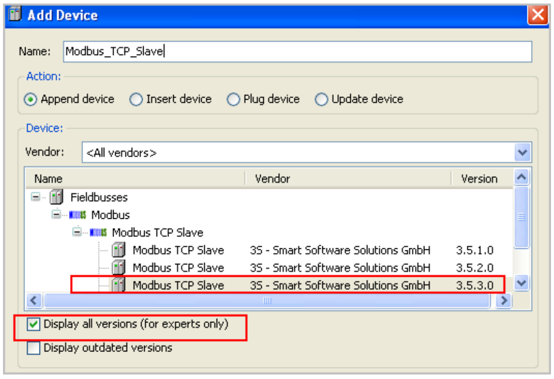
Name	Vendor	Version
Fieldbusses		
EthernetIP		
Modbus		
Modbus TCP Master		
Modbus TCP Master	3S - Smart Software Solutions GmbH	3.5.3.0
ModbusTCP Slave Device		

□ Display all versions (for experts only)

□ Display outdated versions

Information:

- Add “Modbus TCP Slave”
  - Mark “Modbus_TCP_Master”, Modbus TCP Slave’/ “Modbus TCP Slave, ‘Add Device’



Name: Modbus_TCP_Slave

Action: ● Append device ○ Insert device ○ Plug device ○ Update device

Device:

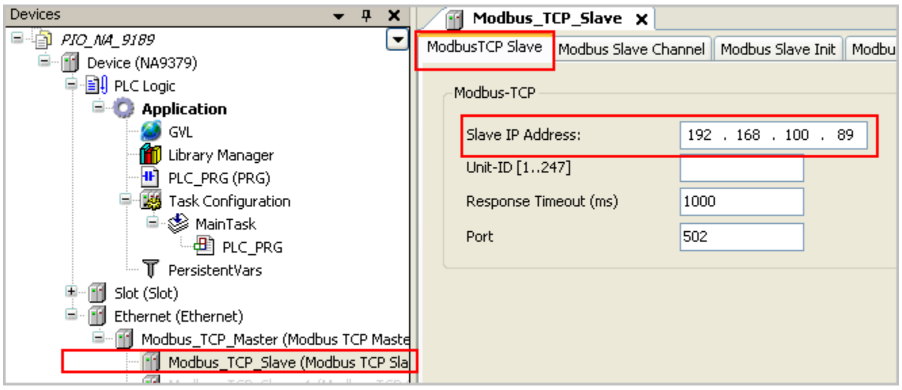
Vendor: <All vendors>

Name	Vendor	Version
Fieldbusses		
Modbus		
Modbus TCP Slave		
Modbus TCP Slave	3S - Smart Software Solutions GmbH	3.5.1.0
Modbus TCP Slave	3S - Smart Software Solutions GmbH	3.5.2.0
Modbus TCP Slave	3S - Smart Software Solutions GmbH	3.5.3.0

☑ Display all versions (for experts only)

□ Display outdated versions

- Configuration the ModbusTCP Slave
  Click ‘Modbus_TCP_Slave’(Modbus TCP Slave) and Write Slave IP Address



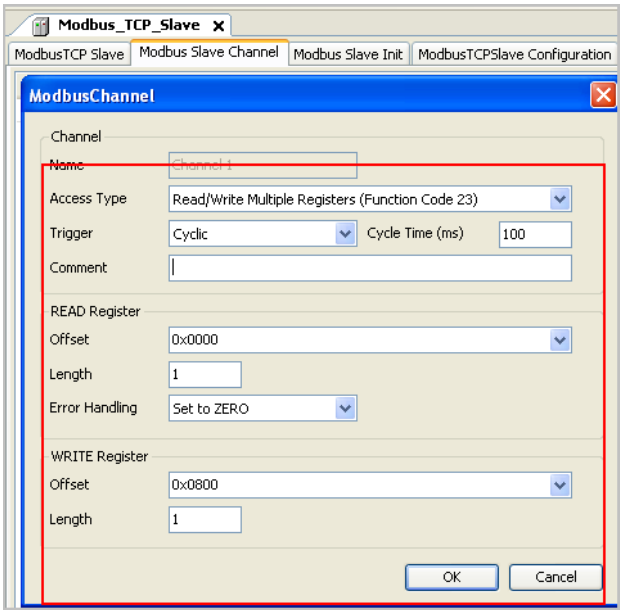
Slave IP Address: 192 . 168 . 100 . 89

Unit-ID [1..247]:

Response Timeout (ms): 1000

Port: 502

- Click ‘Modbus Slave Channel’
  Add a channel of NA-9379 and Modify Cycle Time



Channel

Name: Channel 1

Access Type: Read/Write Multiple Registers (Function Code 23)

Trigger: Cyclic Cycle Time (ms): 100

Comment:

READ Register

Offset: 0x0000

Length: 1

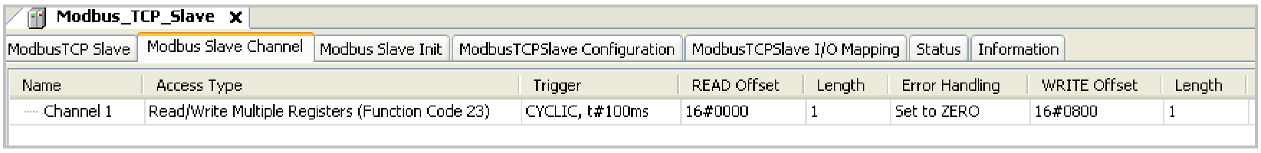
Error Handling: Set to ZERO

WRITE Register

Offset: 0x0800

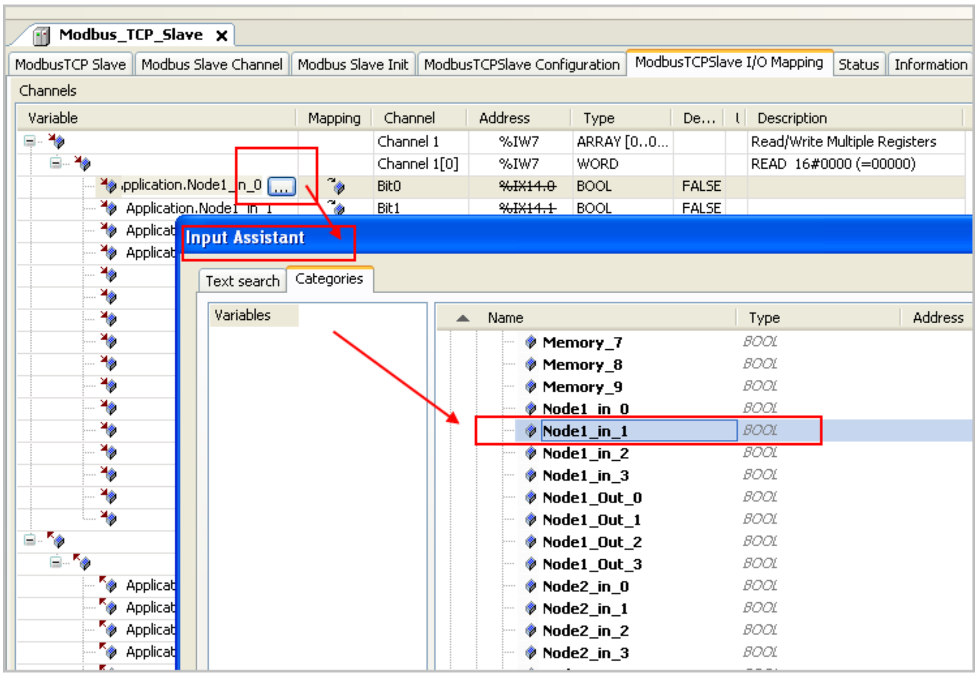
Length: 1

** DefaultCycle Time(ms): 100ms.*



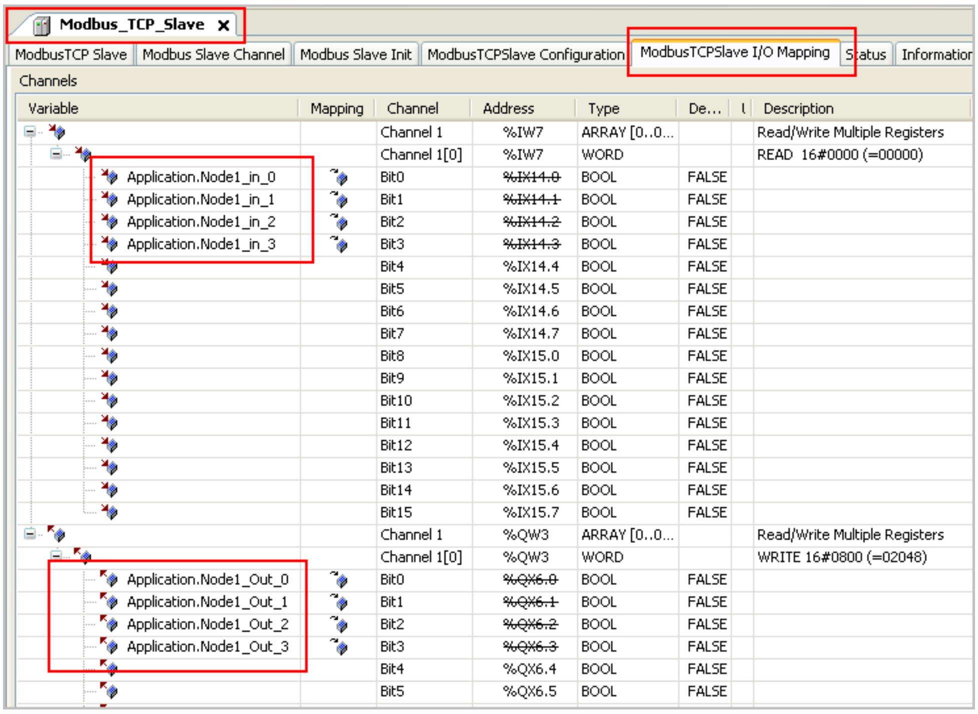
Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length
Channel 1	Read/Write Multiple Registers (Function Code 23)	CYCLIC, t#100ms	16#0000	1	Set to ZERO	16#0800	1

- Connect Modbus TCP Slave I/O to global variable



Variable	Mapping	Channel	Address	Type	De...	U	Description
		Channel 1	%IW7	ARRAY [0..0...			Read/Write Multiple Registers
		Channel 1[0]	%IW7	WORD			READ 16#0000 (=00000)
pplication.Node1_in_0 ...		Bit0	~~%IX14.0~~	BOOL	FALSE		
Application.Node1_in_1		Bit1	~~%IX14.1~~	BOOL	FALSE		
Applicat							
Applicat							
Applicat							
Applicat							
Applicat							
Applicat							

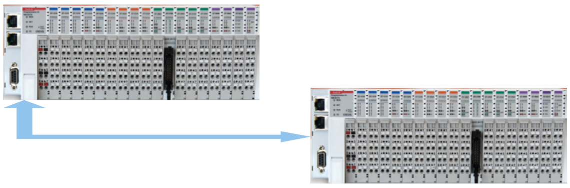
Name	Type	Address
Memory_7	BOOL	
Memory_8	BOOL	
Memory_9	BOOL	
Node1_in_0	BOOL	
Node1_in_1	BOOL	
Node1_in_2	BOOL	
Node1_in_3	BOOL	
Node1_Out_0	BOOL	
Node1_Out_1	BOOL	
Node1_Out_2	BOOL	
Node1_Out_3	BOOL	
Node2_in_0	BOOL	
Node2_in_1	BOOL	
Node2_in_2	BOOL	
Node2_in_3	BOOL	



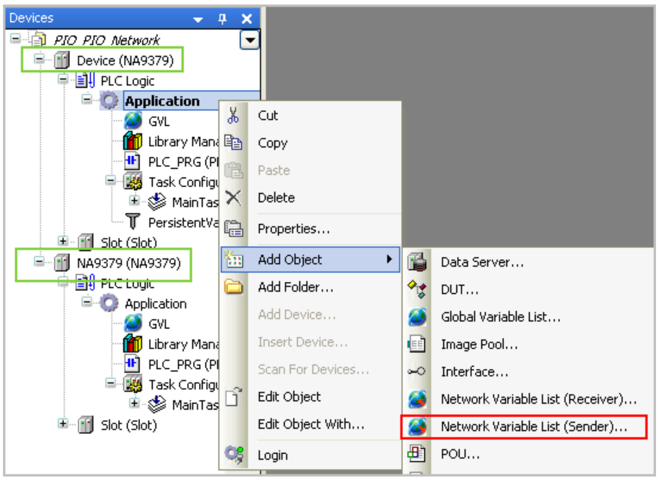
Variable	Mapping	Channel	Address	Type	De...	U	Description
		Channel 1	%IW7	ARRAY [0..0...			Read/Write Multiple Registers
		Channel 1[0]	%IW7	WORD			READ 16#0000 (=00000)
Application.Node1_in_0		Bit0	~~%IX14.0~~	BOOL	FALSE		
Application.Node1_in_1		Bit1	~~%IX14.1~~	BOOL	FALSE		
Application.Node1_in_2		Bit2	~~%IX14.2~~	BOOL	FALSE		
Application.Node1_in_3		Bit3	~~%IX14.3~~	BOOL	FALSE		
		Bit4	%IX14.4	BOOL	FALSE		
		Bit5	%IX14.5	BOOL	FALSE		
		Bit6	%IX14.6	BOOL	FALSE		
		Bit7	%IX14.7	BOOL	FALSE		
		Bit8	%IX15.0	BOOL	FALSE		
		Bit9	%IX15.1	BOOL	FALSE		
		Bit10	%IX15.2	BOOL	FALSE		
		Bit11	%IX15.3	BOOL	FALSE		
		Bit12	%IX15.4	BOOL	FALSE		
		Bit13	%IX15.5	BOOL	FALSE		
		Bit14	%IX15.6	BOOL	FALSE		
		Bit15	%IX15.7	BOOL	FALSE		
		Channel 1	%QW3	ARRAY [0..0...			Read/Write Multiple Registers
		Channel 1[0]	%QW3	WORD			WRITE 16#0800 (=02048)
Application.Node1_Out_0		Bit0	~~%QX6.0~~	BOOL	FALSE		
Application.Node1_Out_1		Bit1	~~%QX6.1~~	BOOL	FALSE		
Application.Node1_Out_2		Bit2	~~%QX6.2~~	BOOL	FALSE		
Application.Node1_Out_3		Bit3	~~%QX6.3~~	BOOL	FALSE		
		Bit4	%QX6.4	BOOL	FALSE		
		Bit5	%QX6.5	BOOL	FALSE		

5.5 PIO to PIO

Communication between PIO and PIO using Network variable

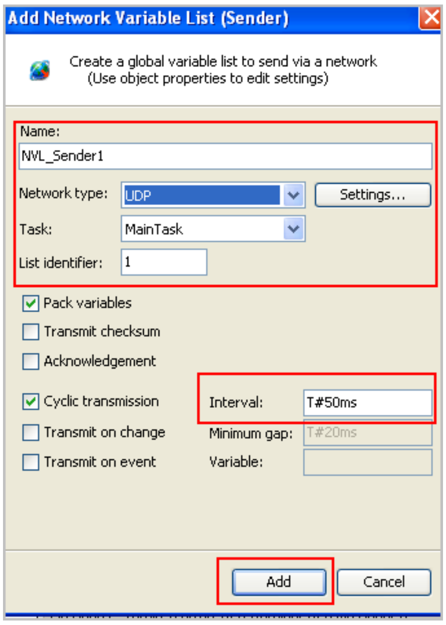


- Click ‘Application’
  - Right click and click ‘Add Object’
  - Click ‘Network Variable List (**Sender**)’.



** You have to add one more devices in the devices tree (marked green)*

- Define the network properties of the sender GVL



Add Network Variable List (Sender)

Create a global variable list to send via a network
(Use object properties to edit settings)

Name: NVL_Sender1

Network type: UDP

Task: MainTask

List identifier: 1

☑ Pack variables

☐ Transmit checksum

☐ Acknowledgement

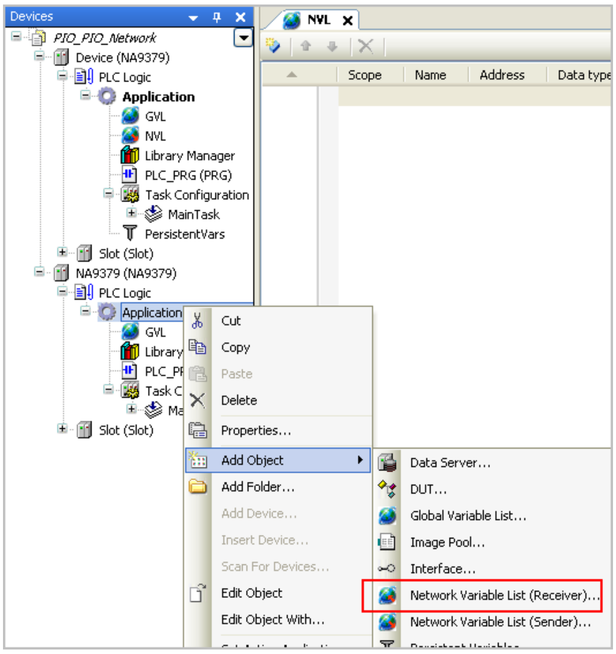
☑ Cyclic transmission — Interval: T#50ms

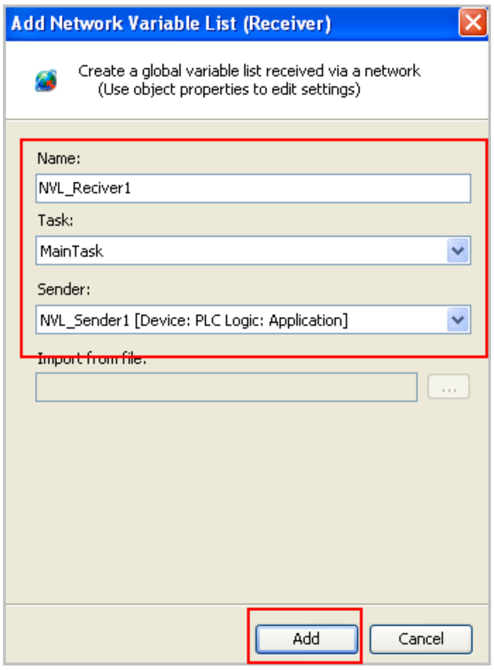
☐ Transmit on change — Minimum gap: T#20ms

☐ Transmit on event — Variable:

You have to select UDP as network type.
** List identifier and Node ID is the same concept.*

- Add a Global Network Variable List in the **Receiver**





Create a global variable list received via a network
(Use object properties to edit settings)

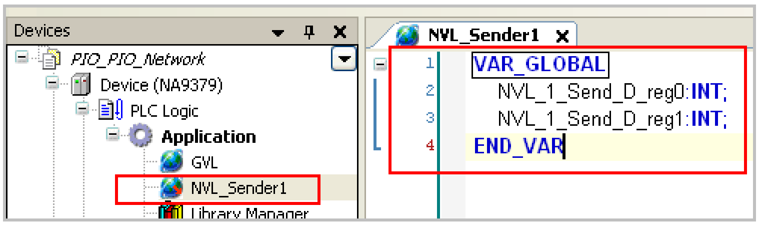
Name: NVL_Reciver1

Task: MainTask

Sender: NVL_Sender1 [Device: PLC Logic: Application]

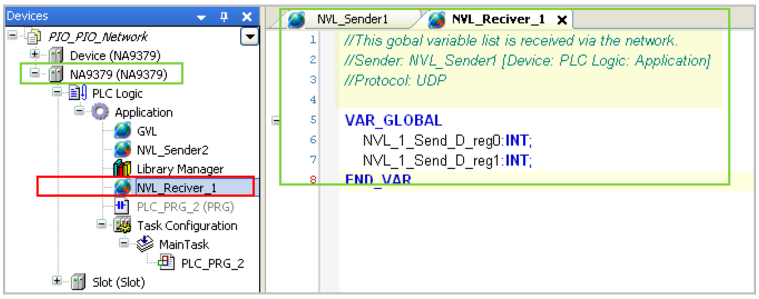
** You find a selection list of all GVLs with network properties currently available in the project.*

- Created by Global Variables



```
VAR_GLOBAL
    NVL_1_Send_D_reg0:INT;
    NVL_1_Send_D_reg1:INT;
END_VAR
```

- Network variable in “Reciver” come from “Sender”

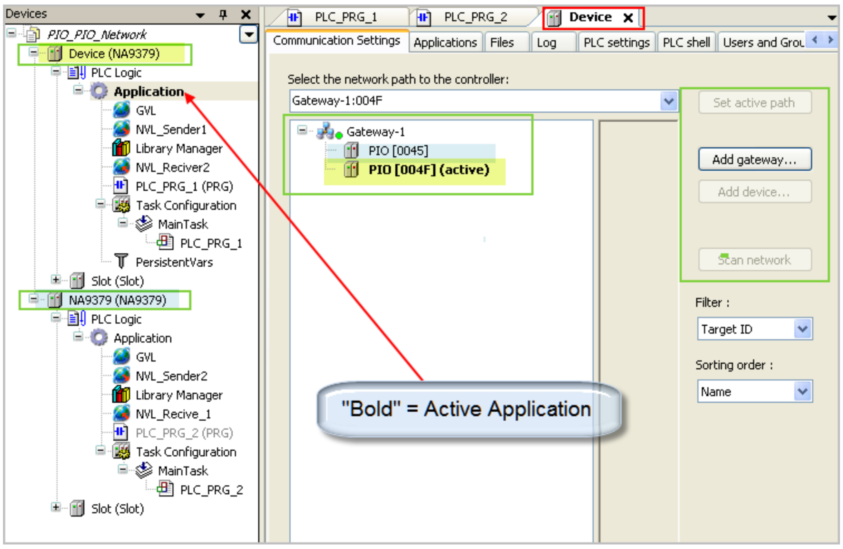


```
//This gobal variable list is received via the network.
//Sender: NVL_Sender1 [Device: PLC Logic: Application]
//Protocol: UDP

VAR_GLOBAL
  NVL_1_Send_D_reg0:INT;
  NVL_1_Send_D_reg1:INT;
END_VAR
```

5.5.1 Download and Monitoring

- Scan network
  After completing the search, double click the Gateway icon to make it activated



Select the network path to the controller: Gateway-1:004F

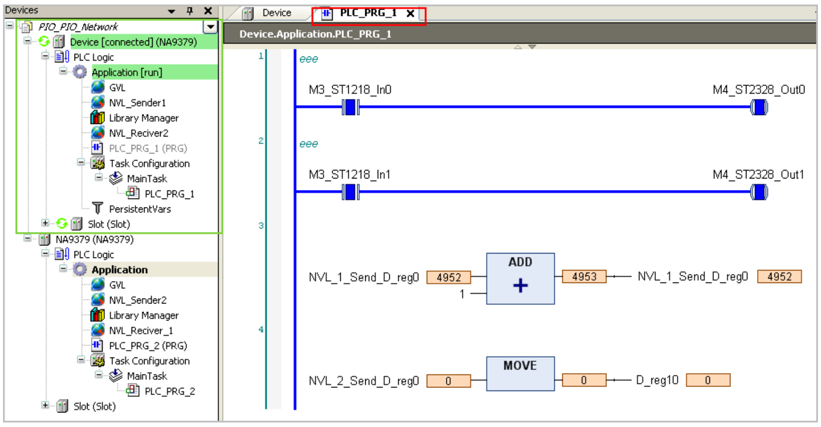
Filter : Target ID

Sorting order : Name

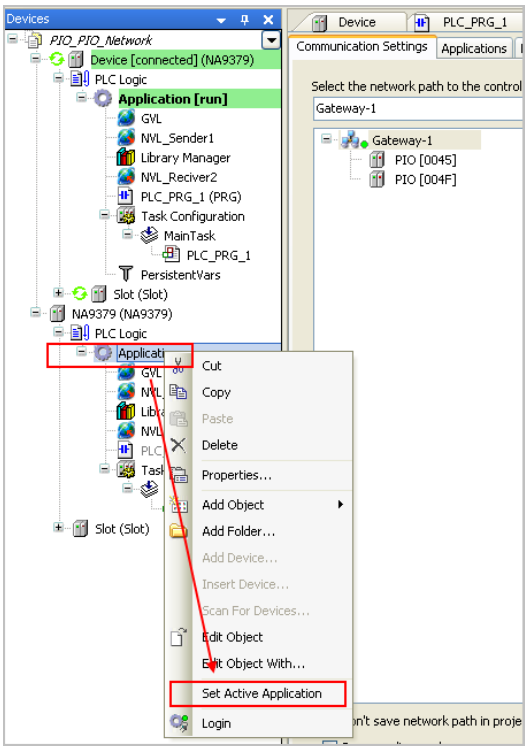
"Bold" = Active Application

- Login (download project)
  [Online], [Login], Download to Application, Entry into Monitoring Mode, [Debug], [RUN]

  PLC nr 1 “Device (NA9379)”

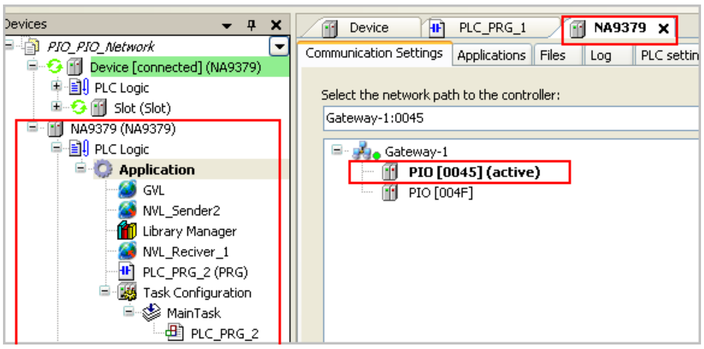


- PLC nr 2 “NA9379(NA9379)”
  Before download to PLC2, set the PLC to “Set Active Application”

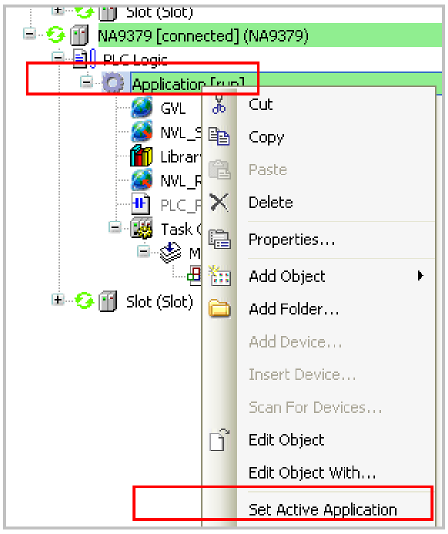


The text “Application” are “BOLD” when active

- Set the other PLCto active and login.



NOTE!
For “Loggin” or “Loggout” to a PLC you have too “Set Active Application” for the PLC



6 Communication interface, MODBUS RTU Slave

More information (detailed information), read the manual: *NA-9379, User Manual*

6.1 Supported function

Function Code	Function	Description
1 (0x01)	Read Coils	Read output bit
2 (0x02)	Read Discrete Inputs	Read input bit
3 (0x03)	Read Holding Registers	Read output word
4 (0x04)	Read Input Registers	Read input word
5 (0x05)	Write Single Coil	Write one bit output
6 (0x06)	Write Single Register	Write one word output
8 (0x08)	Diagnostics	Read diagnostic register
15 (0x0F)	Write Multiple Coils	Write a number of output bits
16 (0x10)	Write Multiple registers	Write a number of output words
23 (0x17)	Read/Write Multiple	Read a number of input words /Write a number of output words

6.2 Specification for function code

6.2.1 Reset Parameter and Erase CODESYS project

MODBUS function code “08” can be used to reset PIO´s IP address, Subnet and gateway to default value and erase the codesys project.

Sub-function 0x0001 (1) Restart Communications Option
The remote device could be initialized and restarted, and all of its communications event counters are cleared. Especially, data field 0x55AA make the remote device to restart with factory default setup of EEPROM.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0001 (1)	0x0000 or 0xFF00	Echo Request Data	Reset
0x0001 (1)	0x55AA	Echo Request Data	Reset with Parameter default *
0x0001 (1)	0x55BB	Echo Request Data	Erase CoDeSys program

* NA-9379 and slot parameter will be the factory defaults value

6.3 Adapter Register Mapping

The special register map can be accessed by function code 3, 4, 6 and 16. Also the special register map must be accessed by read/write of every each address (one address).

Address (hex)	IEC Address (decimal)	Contents
0x0000~0x027F	%IW0~%IW639	640 word Input and Internal memory (Area is write-protected)
0x0280~0x07FF	-	Illegal data address
0x0800~0x0A7F	%QW0~%QW639	640 word Output and Internal memory (Area is write-enabled)
0x0A80~0x0FFF	-	Illegal data address
0x1000~0x1FFF	-	Special Function Register (PIO Information)
0x2000~0x2FFF	-	Special Function Register (Slot Information)
0x4000~0x427F	%MW0~%MW639	640 word Internal memory (Area is write-enabled)

6.4 Adapter Identification Special Resgister (0x1000, 4096)

Address	Access	Type, Size	Description
0x1000 (4096)	Read	1 word	Vendor ID = 0x02E5 (741), Crevis. Co, Ltd.
0x1001 (4097)	Read	1 word	Device type = 0x000C, Network Adapter
0x1002 (4098)	Read	1 word	Product Code = 0x2000
0x1003 (4099)	Read	1 word	Firmware revision, if 0x0101, revision 1.01
0x1004 (4100)	Read	2 words	Product unique serial number
0x1005 (4101)	Read	String up to 34byte	Product name string First 1word is length of valid character string Example) response as following “00 1D 52 4E 2D 39 32 32 32 2C 50 72 6F 66 69 62 75 73 20 41 64 61 70 74 65 72 2C 52 42 55 53 00 00 000” Valid character size = 0x0017 =29 characters "NA-9379,ModbusTCP,PIO,Fn-bus"
0x1006 (4102)	Read	1 word	Sum check of EEPROM
0x1010 (4112)	Read	2 words	Firmware release date
0x1011 (4113)	Read	2 words	Product manufacturing inspection date
0x1012 (4114)	Read	String up to 34byte	Vendor name string First 1word is length of valid character string.
0x101E (4126)	Read	15words 7words	Composite Id of following address 0x1050(4176),0x1051(4177),0x1052(4178),0x1053(4179), 0x1000(4096),0x1001(4097),0x1002(4098),0x1003(4099), 0x1004(4100)

- String Type consists of valid string length (first 1word) and array of characters

6.5 Adapter Watchdog Time and Time Special Register (0x1020, 4128)

A watchdog timer can be configured for timeout periods up to 65535(1unit=100msec). The Watchdog timer will timeout (timer decreased, reached 0) if ModBus operation to the slave node does not occur over the configured watchdog value, then the slave adapter forces that slot output value is automatically set to user-configured fault actions and values.

Address	Access	Type,	Description
0x1020(4128)	Read/ Write	1word	Watchdog time value 16bit unsigned. The time value is represented by multiples of 100msec. The 0 (watchdog timeout disabled) is default value. A changing of watchdog time value resets watchdog error and counter.
0x1021(4129)	Read	1word	Watchdog timer remain value this value is decreased every 100msec
0x1022(4130)	Read	1word	Watchdog error counter, it is cleared by writing address 0x1020
0x1023(4131)	Read/ Write	1word	Enable/disable auto recovery Watchdog error when receiving new frame. 0:Disable, 1:Enable(default). Its value is stored in EEPROM.
0x1028(4136)	Read	4words	I/O update time, FnBus Process time, CoDeSys update time, CoDeSys Process Time. (1usec unit)

6.6 Adapter Information, Special Register (0x1100, 4352)

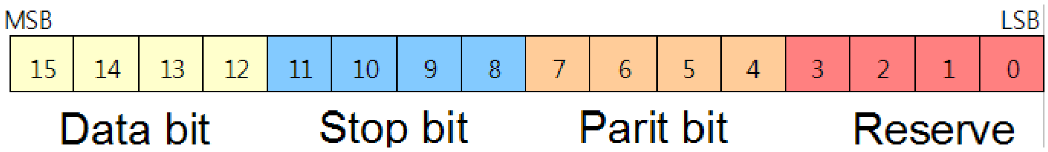
Address	Access	Type, Size	Description
0x1100(4352)			Reserved.
0x1101(4353)			Reserved.
0x1102(4354)	Read	1word	Start address of input image word register. =0x0000
0x1103(4355)	Read	1word	Start address of output image word register. =0x0800
0x1104(4356)	Read	1word	Size of input image word register.
0x1105(4357)	Read	1word	Size of output image word register.
0x1106(4358)	Read	1word	Start address of input image bit. = 0x0000
0x1107(4359)	Read	1word	Start address of output image bit. =0x1000
0x1108(4360)	Read	1word	Size of input image bit.
0x1109(4361)	Read	1word	Size of output image bit.
0x110D(4365)	Read/Write	1word	Enable/Disable Auto Reboot when FnBus, 0:Disable(Default)
0x110E(4366)	Read	Up to 33word	Expansion slot's ST-number including NA First 1word is adapter's number, if NA-9379, then 0x9379
0x1110(4368)	Read	1word	Number of expansion slot
0x1111(4369)	Read	1word	Number of active slot
0x1112(4370)	Read	1word	Number of inactive slot
0x1113(4371)	Read	Up to 33word	Expansion slot Module Id. First 1word is adapter's module id.
0x1114(4372)*	Read/Write	1word	Input process image mode. The default value is 2. Valid value range is from 0 to 3.
0x1115(4373)*	Read/Write	1word	Output process image mode. The default value is 0. Valid value range is from 0 to 1.
0x1116(4374) **	Read/Write	2words	Inactive slot list, The corresponding bit represents slot position. 0:Active slot, 1:Inactive slot. Ex) if value is 0x0001, 0x8000, then slot#1,#32 are inactive
0x1117(4375)	Read	2words	Live slot list , The corresponding bit represents slot position. 1:live slot, 0:not live slot
0x1118(4376)	Read	2words	Alarm slot list. The corresponding bit represents slot position. 1:Alarm slot, 0:Normal slot
0x1119(4377)	Read	1word	Hi byte is ModBus status, low byte is FnBus status. It is identical with address 0x1040.
0x111A(4378)	Write	1word	Reserved. Adapter Scan command.
0x111B(4379)	Read/Write	1word	Reserved. IO State machine.
0x111C(4380)	Read	2words	Reserved. Runtime fault code.
0x111D(4381)	Read	1word	Adapter FnBus Revision. If 0x013C, FnBus Revision is 1.60
0x111E(4382)	Read	1word	Reserved. Adapter IO identification vendor code.
0x111F(4383)	Read	5words	LED Display Value and Status Code

* After the system is reset, the new "Set Value" action is applied.
** If the slot location is changed, set default value automatically (all expansion slots are live).

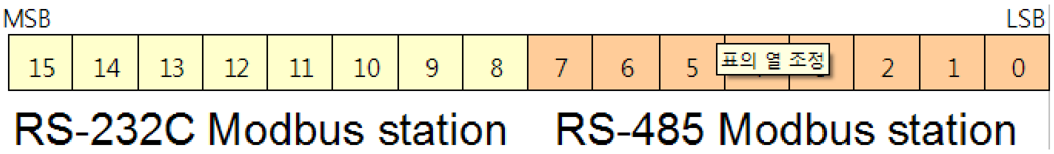
6.7 Adapter Setting Special Register (0x1600, 5632)

Address	Access	Type, Size	Description
0x1600(5632)	Read	2 word	IP Address (ex : C0A8 0005 = 192.168.0.5)
0x1602(5634)	Read	2 word	Subnet Mask (ex : FFFF FF00 = 255.255.255.0)
0x1604(5636)	Read	2 word	Gateway (ex : C0A8 0001 = 192.168.0.1)
0x1606(5638)	Read/Write	1 word	RS-232C Baud rate (1200bps~115200bps) 0 : 1200 4 : 19200 1 : 2400 5 : 38400 (default) 2 : 4800 6 : 57600 3 : 9600 7 : 115200
0x1607(5639)	Read/Write	1 word	*RS-232C Setting. 1 nibble : Data bit (0 : 8bit (default), 1 : 9 bit) 2 nibble : Stop bit (0 : 1bit (default), 1 : 2 bit) 3 nibble : Parity bit (0 : none (default), 1: even, 2 : odd) 4 nibble : Reserve
0x1608(5640)	Read/Write	1 word	RS-485 Baud rate. (1200bps~115200bps) 0 : 1200 4 : 19200 1 : 2400 5 : 38400 (default) 2 : 4800 6 : 57600 3 : 9600 7 : 115200
0x1609(5641)	Read/Write	1 word	RS-485 Setting. 1 nibble : Data bit (0 : 8bit (default), 1 : 9 bit) 2 nibble : Stop bit (0 : 1bit (default), 1 : 2 bit) 3 nibble : Parity bit (0 : none (default), 1: even, 2 : odd) 4 nibble : Reserve
0x160A(5642)	Read/Write	1 word	**Modbus Station. High 1byte : Station No. of RS-232C (default : 1) Low 1byte : Station No. of RS-485 (default : 1)
0x160B(5643)	Read/Write	1 word	IP Setting Mothod. - Not use : 0x0000 - BootP : 0x8000 (default) - DHCP : 0x8001
0x1610(5648)	Read	3 word	Mac Address (ex: 0014 F700 0101 = 00.14.F7.00.01.01)
0x1620(5664)	Read/Write	4 word	RTC 1 word : 00ss (ss : sec) 2 word : hhmm (hh : hour, mm : min) 3 word : mmdd (mm : month, dd : day) 4 word : yyyy (yyyy : year) (ex : 07D8 0514 0F19 0006 = 2008. 05.20. 15.25. 06)

***RS-232C Setting :** This description for 0x1607/0x1609 register with bit.



****Modbus Station :** This description for 0x160A register with bit.


MSB
LSB
15
14
13
12
11
10
9
8
7
6
5
표의 열 조정
2
1
0
RS-232C Modbus station
RS-485 Modbus station